

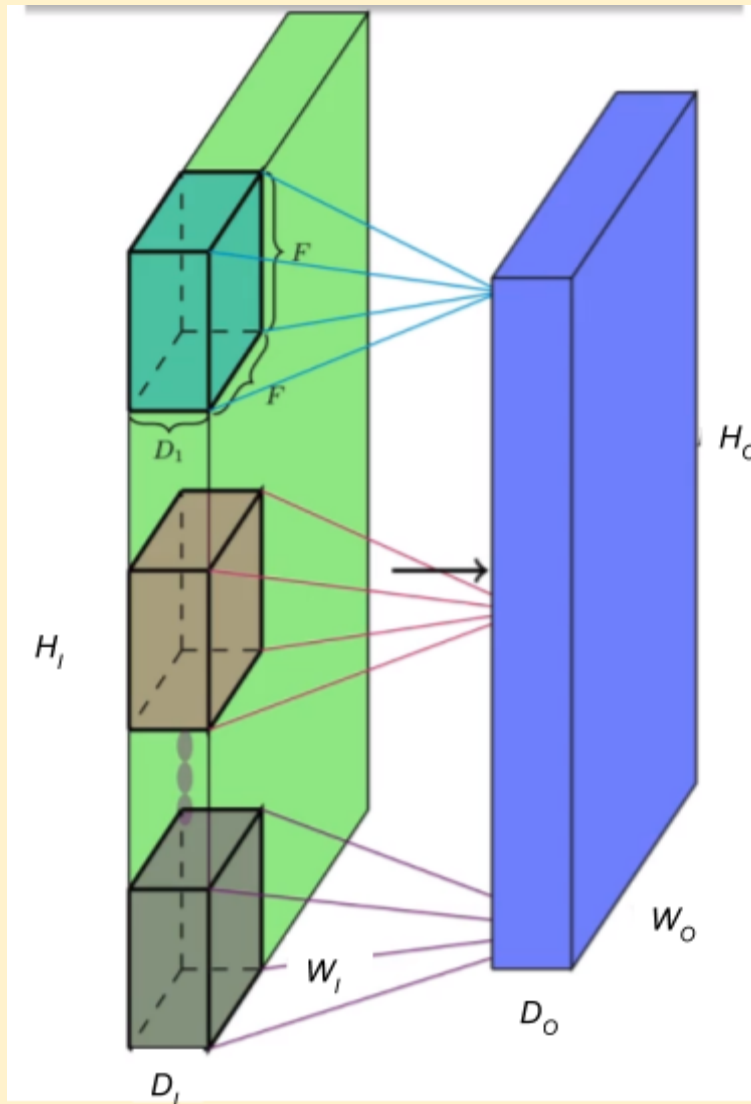
# PadhAI: The Convolution Operation

## One Fourth Labs

### Terminology

Let's look at some terminology

1. Consider the following 3D convolution operation and look at the terminology associated with it



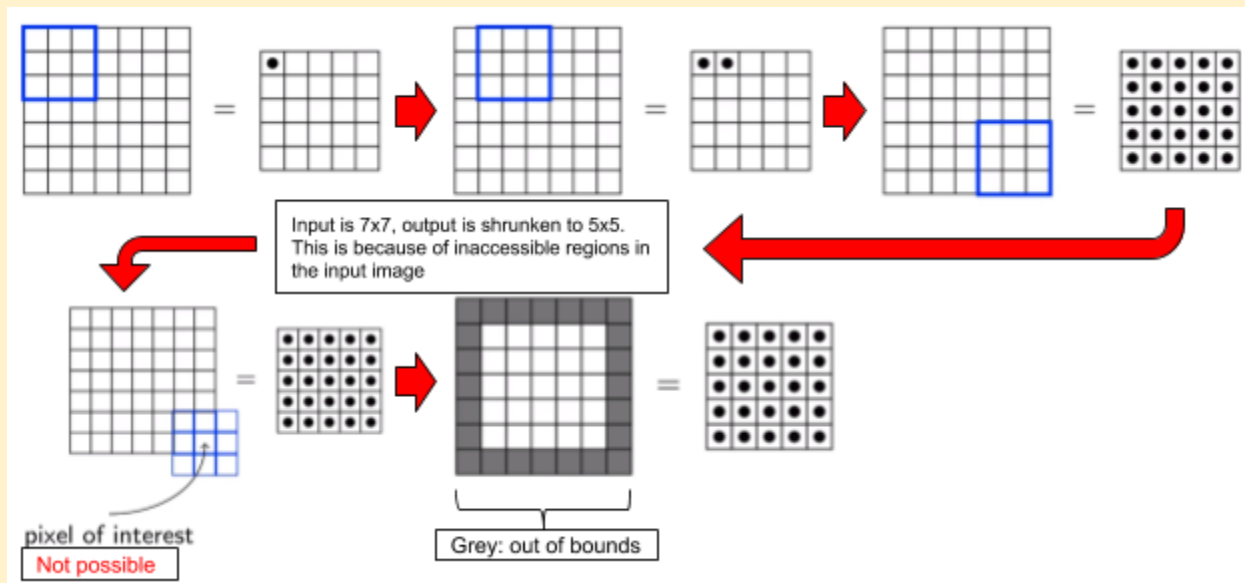
### 2. Terminology:

- a. Input Width ( $W_i$ ), Height ( $H_i$ ) and Depth ( $D_i$ )
  - b. Output Width ( $W_o$ ), Height ( $H_o$ ) and Depth ( $D_o$ )
  - c. The spatial extent of a filter ( $F$ ), a single number to denote width and height as they are equal
  - d. Filter depth is always the same as the Input Depth ( $D_i$ )
  - e. The number of filters ( $K$ )
  - f. Padding ( $P$ ) and Stride ( $S$ )
3. **Question:** Given  $W_i$ ,  $H_i$ ,  $D_i$ ,  $F$ ,  $K$ ,  $S$  and  $P$  how do you compute  $W_o$ ,  $H_o$ , and  $D_o$ ?

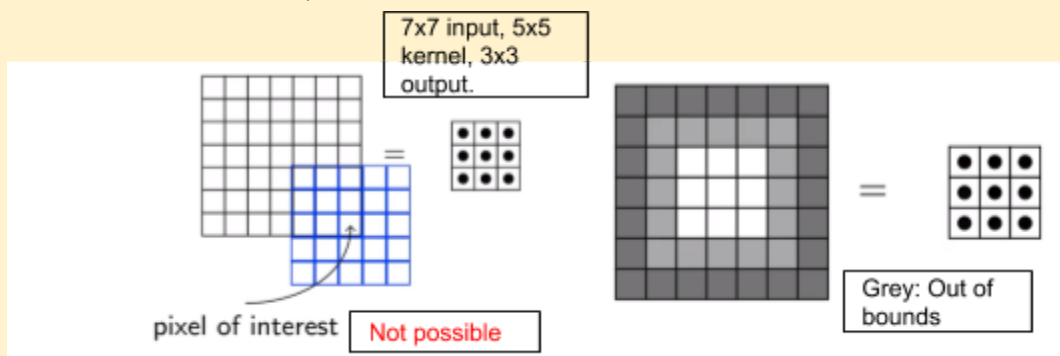
# PadhAI: The Convolution Operation

## One Fourth Labs

4. To answer that, let us look at a sample 3x3 kernel passing over a 7x7 image



- Here, we can see that by running the 3x3 kernel over a 7x7 image, we get a smaller 5x5 image.
  - This is because we can't place the kernel at the corners as it will cross the input boundaries
  - This is true of all the shaded points.
  - Hence the size of the output will be smaller than that of the input
5. Let's see another example with a 5x5 kernel



- Here, we can see that by running a 5x5 kernel over a 7x7 input, we get a smaller 3x3 image
  - Here, the out-of-bounds regions are larger.
  - Thus the output is much smaller.
6. We can see that the reduction in size can be given by the following equations
- $W_O = W_I - F + 1$
  - $H = H_I - F + 1$
7. However in practice, we could still place the kernel on the boundary and take only the valid neighbors. This is roughly what is being done.